

Checking EE Memory Accesses on the DTL-T10000

To: All licensees

2000/01/28
SCE Software Development Division

The following is a description of how accesses to EE memory above 32 Mbytes can be checked on the DTL-T10000.

If a program does not work properly on the DTL-H10000 debugging station, checking for such accesses can be useful in debugging.

* Procedure

- 1) Edit lines 1 and 2 in /usr/local/sce/ee/lib/app.cmd, then build so that the stack fits within the 32M limit.

```
(Example)
_stack_size = 0x00100000;
_stack = 0x01f00000;
```

- 2) Use the hardware breakpoint feature of dsedb to set a breakpoint if access is attempted in the 32M - 64M area.

```
(Example)
$ dsedb
:
dsedb S> hbp dausk:02000000,d2000000
dsedb S> hbp pcusk:02000000,d2000000
dsedb S> run main.elf
```

- 3) If access to the 32 - 64M area is attempted, the following will be output and a breakpoint will occur.

```
*** Debug Exception
at=22000000 v0-1=00000004,00000000 a0-3=...
:
$cause   = 0x00038000 [ CE0 EXC2=Debug IP7 EXC="External Interrupt" ]
$status  = 0x70030c15 [ Cu210 EDI EIE IM3 IM2 KSU=User ERL IE ]
0x002000ec: 0x3c040021 lui      $a0,0x0021
0x002000f0: 0x3c012200 lui      $at,0x2200 # 0x22000000
->0x002000f4: 0xac300000 sw       $s0,0($at)
:
```

- 4) Then set breakpoints as described above in 2) to catch accesses in the 64M - 128M area.

```
(Example)
$ dsedb
:
dsedb S> pload main.elf
:
dsedb S> hbp dausk:04000000,d4000000
```

```
dsedb S> hbp pcusk:04000000,d4000000
dsedb S> run
```

* Addendum

Hardware breakpoints will no longer be in effect after an IOP reboot, so the procedure described above cannot be used when default modules are replaced. To enable hardware breakpoints after an IOP reboot, the program must be stopped and hbp must be used as described below.

- 1) Enter a break command at any time, set hbp(s), then continue.

```
(Example)
:
dsedb R> break
:
dsedb S> hbp dausk:04000000,d4000000
dsedb S> hbp pcusk:04000000,d4000000
dsedb S> cont
```

- 2) Insert a break instruction in the source after the IOP reboot, at an arbitrary position and build.
After the break, set hbp(s) and continue.

```
(Example: C source)
asm("break");
```

```
(Example: dsedb)
dsedb S> run main.elf
Loading program (address=0x00200000 size=0x0000c6c6) ...
:
$cause   = 0x00000024 [ CE0 EXC2=Reset EXC="Breakpoint" ]
$status  = 0x70030c13 [ Cu210 EDI EIE IM3 IM2 KSU=User EXL IE ]
0x002000cc: 0x0c0800a6 jal      0x00200298 # <__main>
0x002000d0| 0x00000000 nop
->0x002000d4: 0x0000000d break    0x0
:
dsedb S> sr $epc $epc+4
dsedb S> hbp dausk:04000000,d4000000
dsedb S> hbp pcusk:04000000,d4000000
dsedb S> cont
```